A High-Speed Asynchronous Decompression Circuit for Embedded Processors

Full text Publisher Site

Source **ARVLSI** archive
Proceedings of the 17th Conference on Advanced Research in VLSI (ARVLSI '97) table of contents
Page: 219
Year of Publication: 1997
ISBN:0-8186-7913-1

Authors Martin Benes
Andrew Wolfe
Steven M. Nowick

Publisher IEEE Computer Society Washington, DC, USA

↑ ABSTRACT

This paper describes the architecture and implementation of a high-speed decompression engine for embedded processors. The engine is targeted to processors where embedded programs are stored in compressed form, and decompressed at runtime during instruction cache refill. The decompression engine uses a unique asynchronous variable decompression rate architecture to process Huffman-encoded instructions. The resulting circuit is significantly smaller than comparable synchronous decoders, yet has a higher throughput rate than almost all existing designs. The 0.8 micron layout is all full-custom and contains predominantly dynamic domino logic. The top-level control, as well as several small state machines, are implemented using asynchronous logic. The design operates without a user-supplied clock. Simulations using Lsim show average throughput of 32 bits/45 ns on the output side, corresponding to about 480 Mbit/sec on the input side. The chip has been manufactured by MOSIS; tests show that the asynchronous implementation operates correctly, with an average throughput exceeding simulations: 32 bits/39 ns on the output side, corresponding to about 560 Mbit/sec on the input side. This speed is acceptable for our application. The area of the design (excluding the pad-frame overhead) is only 0.75~\hbox{mm}^2. The design is the first fabricated chip for an instruction decompression unit for embedded processors.

↑ CITINGS *5*

Charles Lefurgy , Eva Piccininni , Trevor Mudge, Evaluation of a high performance code compression method, Proceedings of the 32nd annual ACM/IEEE international symposium on Microarchitecture, p.93-102, November 16-18, 1999, Haifa, Israel

Kelvin Lin , Chung-Ping Chung , Jean Jyh-Jiun Shann, Compressing MIPS code by multiple operand dependencies, ACM Transactions on Embedded Computing Systems (TECS), v.2 n.4, p.482-508,

November 2003

Árpád Beszédes , Rudolf Ferenc , Tibor Gyimóthy , André Dolenc , Konsta Karsisto, Survey of code-size reduction methods, ACM Computing Surveys (CSUR), v.35 n.3, p.223-267, September 2003

Darko Kirovski , Johnson Kin , William H. Mangione-Smith, Procedure based program compression, Proceedings of the 30th annual ACM/IEEE international symposium on Microarchitecture, p.204-213, December 01-03, 1997, Research Triangle Park, North Carolina, United States

Sergei Y. Larin , Thomas M. Conte, Compiler-driven cached code compression schemes for embedded ILP processors, Proceedings of the 32nd annual ACM/IEEE international symposium on Microarchitecture, p.82-92, November 16-18, 1999, Haifa, Israel

↑ **Collaborative Colleagues:**

Martin Benes:	Arthur Abnous Varghese George Steven M. Nowick Vandana Prabhu Jan Rabaey Marlene Wan Andrew Wolfe Hui Zhang			
Steven M. Nowick:	Peter A. Beerel Martin Bene_ Martin Benes Tiberiu Chelcea Bill Coates Mark E. Dean David L. Dill Ayoob E. Dooply Robert M. Fuhrer Robert Mack Fuhrer	Ganesh Gopalakrishnan Mark Horowitz Hans Jacobson Prabhakar Kudva Luciano Lavagno Bill Lin Charles W. O 'Donnell Massoud Pedram Sergey Rylov Alexander Rylyakov	Montek Singh Robert F. Sproull Michael Theobald Jose A. Tierno Andrew Wolfe Tao Wu Pei-Chuan Yeh Kenneth Y. Yun	
Andrew Wolfe:	Martin Bene_ Martin Benes Rodney Boleyn Yin Chan Alex Chanin Steve Chinatti Thomas M. Conte Pradeep K. Dubey Santanu Dutta Edil S. Tavares Fernandes Eliseu M. Chaves Filho	Matthew D. Jennings Ravi Koshy Michael Kozuch John Kubiatowicz Mike Tien-Chien Lee Ruby B. Lee Yau-Tsun Steven Li Heng Liao Bede Liu Sharad Malik William Mangione-Smith	Daniel Markham Don McKay Derek B. Noonburg Steven M. Nowick Alex Peleg Salliah Rathnam Michael Schlansker Mike Schlansker John P. Shen Gary Slater Peter Song	Ashok Sudarsanam Spencer Sun Edil S. Tavares Fernandes Vivek Tiwari Wayne Wolf Boon-Lock Yeo Minerva M. Yeung Alberto Ferreira de Souza

Terms used **asynchronous huffman** Found **5,995** of **141,345**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 Session 6C: Markovian analysis and asynchronous circuits: Pipeline optimization for asynchronous circuits: complexity analysis and an efficient optimal algorithm

Sangyun Kim, Peter A. Beerel

November 2000 **Proceedings of the 2000 IEEE/ACM international conference on Computer-aided design**

Full text available: pdf(113.08 KB) Additional Information: full citation, abstract, references

This paper addresses the problem of identifying the minimal pipelining needed in an asynchronous circuit (e.g., number/size of pipeline stages/latches required) to satisfy a given performance constraint, thereby implicitly minimizing area and power for a given performance. In contrast to the somewhat analogous problem of *retiming* in the synchronous domain, we first show that the basic pipeline optimization problem for asynchronous circuits is NP-complete. This paper then presents an effic ...

2 Statistically optimized asynchronous barrel shifters for variable length codecs

Peter A. Beerel, Sangyun Kim, Pei-Chuan Yeh, Kyeounsoo Kim

August 1999 **Proceedings of the 1999 international symposium on Low power electronics and design**

Full text available: pdf(337.23 KB) Additional Information: full citation, references, index terms

3 Generation of fast interpreters for Huffman compressed bytecode

Mario Latendresse, Marc Feeley

June 2003 **Proceedings of the 2003 workshop on Interpreters, Virtual Machines and Emulators**

Full text available: pdf(323.22 KB) Additional Information: full citation, abstract, references, index terms

Embedded systems often have severe memory constraints requiring careful encoding of programs. For example, smart cards have on the order of 1K of RAM, 16K of non-volatile memory, and 24K of ROM. A virtual machine can be an effective approach to obtain compact programs but instructions are commonly encoded using one byte for the opcode and multiple bytes for the operands, which can be wasteful and thus limit the size of programs runnable on embedded systems. Our approach uses canonical Huffman co ...

Keywords: Java, canonical Huffman code, code compression, decoder

4 Optimization: Reducing probabilistic timed petri nets for asynchronous architectural analysis

Sangyun Kim, Sunan Tugsinavisut, Peter Beerel

December 2002 **Proceedings of the 8th ACM/IEEE international workshop on Timing issues in the specification and synthesis of digital systems**

Full text available: pdf(541.04 KB) Additional Information: full citation, abstract, references, index terms

This paper introduces structural reductions of probabilistic timed Petri nets that preserve a large class of performance measurements. In particular, the paper proposes a class of reductions that preserve efficiently computable bounds of statistics of time-separation of events (TSEs). It identifies two specific reductions within this class. It demonstrates the utility of these reductions by reducing a detailed Petri net describing the four-phase protocol of a well-known asynchronous pipeline tem ...

5 Advances in synthesis: Implementing asynchronous circuits using a conventional EDA tool-flow

Christos P. Sotiriou

June 2002 **Proceedings of the 39th conference on Design automation**

Full text available: pdf(107.01 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents an approach by which asynchronous circuits can be realised with a conventional EDA tool flow and conventional standard cell libraries. Based on a gate-level asynchronous circuit implementation technique, direct-mapping, and by identifying the delay constraints and exploiting certain EDA tool features, this paper demonstrates that a conventional EDA tool flow can be used to describe, place, route and timing-verify asynchronous circuits.

Keywords: EDA, asynchronous, tool-flow

6 Practical advances in asynchronous design and in asynchronous/synchronous interfaces

Erik Brunvand, Steven Nowick, Kenneth Yun

June 1999 **Proceedings of the 36th ACM/IEEE conference on Design automation**

Full text available: pdf(155.17 KB) Additional Information: full citation, references, citings, index terms

7 Unifying synchronous/asynchronous state machine synthesis

Kenneth Y. Yun, David L. Dill

November 1993 **Proceedings of the 1993 IEEE/ACM international conference on Computer-aided design**

Full text available: pdf(800.19 KB) Additional Information: full citation, references, citings

8 Testing redundant asynchronous circuits by variable phase splitting

Luciano Lavagno, Antonio Lioy, Michael Kishinevsky

September 1994 **Proceedings of the conference on European design automation**

Full text available: pdf(700.22 KB) Additional Information: full citation, references, citings, index terms

9 Algorithms for synthesis of hazard-free asynchronous circuits

L. Lavagno, K. Keutzer, A. Sangiovanni-Vincentelli

June 1991 **Proceedings of the 28th conference on ACM/IEEE design automation**

Full text available: pdf(868.37 KB) Additional Information: full citation, references, citings, index terms

10 A unified framework for race analysis of asynchronous networks

J. A. Brzozowski, C.-J. Seger

January 1989 **Journal of the ACM (JACM)**, Volume 36 Issue 1

Full text available: pdf(2.11 MB) Additional Information: full citation, abstract, references, citings, index terms, review

A unified framework is developed for the study of asynchronous circuits of both gate and MOS type. A basic network model consisting of a directed graph and a set of vertex excitation functions is introduced. A race analysis model, using three values (0, 1, and x), is developed for studying state transitions in the network. It is shown that the results obtained using this model are equivalent to those using ternary simulation. It is also proved that the set of state variables can be reduced ...

11 An efficient critical race-free state assignment technique for asynchronous finite state machines

Tam Anh Chu, Narayana Mani, Clement K. C. Leung

July 1993 **Proceedings of the 30th international conference on Design automation**

Full text available: pdf(584.54 KB) Additional Information: full citation, references, index terms

12 Survey of code-size reduction methods

Árpád Beszédes, Rudolf Ferenc, Tibor Gyimóthy, André Dolenc, Konsta Karsisto

September 2003 **ACM Computing Surveys (CSUR)**, Volume 35 Issue 3

Full text available: pdf(443.89 KB) Additional Information: full citation, abstract, references, index terms

Program code compression is an emerging research activity that is having an impact in several production areas such as networking and embedded systems. This is because the reduced-sized code can have a positive impact on network traffic and embedded system costs such as memory requirements and power consumption. Although code-size reduction is a relatively new research area, numerous publications already exist on it. The methods published usually have different motivations and a variety of appli ...

Keywords: code compaction, code compression, method assessment, method evaluation

13 Compressing MIPS code by multiple operand dependencies

Kelvin Lin, Chung-Ping Chung, Jean Jyh-Jiun Shann

November 2003 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 2 Issue 4

Full text available: pdf(576.31 KB) Additional Information: full citation, abstract, references, index terms

Intuitively, destination registers of some instructions have great possibilities to be used as the source registers of the immediately subsequent instructions. Such destination register/source register pairs have been exploited previously to improve code compression ratio [*compression ratio* = (*Dictionary Size* + *Encoded Program Size*)/*Original Program Size*]. This paper further examines the exploitation of both register and immediate operand dependencies to improve the c ...

Keywords: Code compression, benchmarks, data compression, instruction set architecture

14 Synthesis of multiple-input change asynchronous finite state machines
Maureen Ladd, William P. Birmingham
June 1991 **Proceedings of the 28th conference on ACM/IEEE design automation**
Full text available: pdf(611.61 KB) Additional Information: full citation, references, index terms

15 Symbolic hazard-free minimization and encoding of asynchronous finite state machines
Robert M. Fuhrer, Bill Lin, Steven M. Nowick
December 1995 **Proceedings of the 1995 IEEE/ACM international conference on Computer-aided design**
Full text available: pdf(147.53 KB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

This paper presents an automated method for the synthesis of multiple-input-change (MIC) asynchronous state machines. Asynchronous state machine design is subtle since, unlike synchronous synthesis, logic must be implemented without hazards, and state codes must be chosen carefully to avoid critical races. We formulate and solve an optimal hazard-free and critical race-free encoding problem for a class of MIC asynchronous state machines called burst-mode. Analogous to a paradigm successfully use ...

Keywords: optimal state assignment, asynchronous state machines, hazards, sequential synthesis, sequential optimization

16 A technique for synthesizing distributed burst-mode circuits
Prabhakar Kudva, Ganesh Gopalakrishnan, Hans Jacobson
June 1996 **Proceedings of the 33rd annual conference on Design automation**
Full text available: pdf(108.03 KB) Additional Information: full citation, references, citings, index terms

17 Memory hierarchies: A code decompression architecture for VLIW processors
Yuan Xie, Wayne Wolf, Haris Lekatsas
December 2001 **Proceedings of the 34th annual ACM/IEEE international symposium on Microarchitecture**
Full text available: pdf(1.00 MB) Publisher Site Additional Information: full citation, abstract, references, citings

In embedded system design, memory has been one of the most restricted resources. Reducing program size has been an important goal when designing an embedded system. Most of the previous work on code compression has targeted RISC architectures. Recently VLIW processors became very popular, particularly for signal processing. Decompression speed is especially important for VLIW architectures given that the length of the instruction word is long. Furthermore, modern VLIW architectures use flexible ...

18 SIGDA 2 - Design automation: Modular requirements for digital logic simulation at a predefined functional level
C. W. Hemming, S. A. Szygenda
August 1972 **Proceedings of the ACM annual conference - Volume 1**
Full text available: pdf(882.95 KB) Additional Information: full citation, abstract, references, citings

Simulation of digital logic provides a viable technique for development and diagnosis of digital systems. Simulation models currently employed are discussed with a summary of

structure and timing techniques. A methodology for functional simulation in conjunction with gate level simulation is discussed, presenting a representative set of predefined functions, and introducing a measure for predefined function performance. Errors in design detectable at the functional level are catagorized.

Keywords: diagnosis of digital systems, digital simulation, fault simulation, functional simulation, logic design

19 Evaluation of a high performance code compression method

Charles Lefurgy, Eva Piccininni, Trevor Mudge

November 1999 **Proceedings of the 32nd annual ACM/IEEE international symposium on Microarchitecture**

Full text available: pdf(1.01 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

Compressing the instructions of an embedded program is important for cost-sensitive low-power control-oriented embedded computing. A number of compression schemes have been proposed to reduce program size. However, the increased instruction density has an accompanying performance cost because the instructions must be decompressed before execution. In this paper, we investigate the performance penalty of a hardware-managed code compression algorithm recently introduced in IBM's PowerPC 405. ...

20 Proof of the equivalent realizability of a time-bounded arbiter and a runt-free inertial delay

B. I. Strom

April 1979 **Proceedings of the 6th annual symposium on Computer architecture**

Full text available: pdf(357.02 KB) Additional Information: full citation, abstract, references, index terms

This paper proves that, given a reliable time-bounded arbiter, it is possible to realize a reliable (i.e. runt-free) inertial delay, and vice-versa. It therefore shows that the time-bounded arbiter and the inertial delay are equally realizable. Consequently all theoretical limitations which apply to one will apply, in some form, to the other as well.

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next